

# Object Recognition for Video Retrieval

Rene Visser, Nicu Sebe, and Erwin Bakker

LIACS Media Lab, Leiden University, Niels Bohrweg 1  
2300 CA Leiden, The Netherlands

**Abstract.** Recognition of objects in video can offer significant benefits to video retrieval including automatic annotation and content based queries based on the object characteristics. This paper describes our preliminary work toward recognizing objects in video sequences and gives a brief survey of the relevant research in the literature. We use the Kalman filter to obtain segmented blobs from the video, classify the blobs using the probability ratio test, and apply several different temporal filtering methods, which results in sequential classification methods over the video sequence containing the blob. Results from real video sequences are shown.

## 1 Introduction

Detecting particular objects in video is an important step toward semantic understanding of visual imagery. For example, in content based retrieval, the ability to detect people and automobiles gives the option of advanced queries such as "Find a video clip which contains a crowded area or a fast moving car."

In this paper, we give a summary of the current work in video segmentation for moving objects followed by the implemented tracking and classification algorithm. The novel aspect of our work is described in Sections 4 and 5 where we use temporal filtering for classifying the moving objects and show some early results. Conclusions are given Section 6.

## 2 Background

The segmentation of moving objects is an important problem in image sequence analysis and in the problem of video retrieval (i.e. [2, 5, 13]). There is significant related research [1-18]. Some approaches toward visual matching apply stochastic parsing [14]; integrate learning [1, 5, 8]; and adaptive background mixtures [7].

A real-time system for tracking people, called Pfinder ("Person finder"), is proposed by Wren, et al. [16]. First, a model of the scene is built by observing the scene when no person is present. For each pixel, the mean color value and the covariance of the associated distribution is determined. Then when a person enters the scene, the system begins to build up a model of that person. This is done by first detecting a large change in the scene, and then building up a multi-blob model of the

person over time. The model building process is driven by the distribution of color on the person's body with additional blobs added for other colored regions.

In the work by Gu, et al., [3], a method for spatio-temporal segmentation of long image sequences of scenes that include multiple independently moving objects is presented. This method is based on the Minimum Description Length (MDL) principle.

A similar application with different techniques has been presented in [6]. In this paper, two approaches for motion based representations are described. The first approach demonstrates that dominant 2D and 3D motion techniques are useful for computing video mosaics through the computation of dominant scene motion and/or structure. However, this may not be adequate if object level indexing and manipulation is to be accomplished efficiently. The second approach addresses this issue through simultaneous estimation of an adequate number of simple 2D motion models. A unified view of the two approaches naturally follows from the multiple model approach: the dominant motion method becomes a particular case of the multiple method if the number of models is fixed to be one and only the robust EM algorithm without the MDL stage is employed.

Another application in the area of videos is presented in [19]. It addresses the problem of automatic video browsing as it describes methods that use edges and motion for detecting production effects, e.g. cuts, fades, dissolves, wipes and captions, and computing motion segmentation. This segmentation involves the computation of the primary and the secondary motion. This is achieved by finding isolated peaks in the error surface formed by the similarity of two images, a scalar function of the displacement that makes the two images most similar. These motions are then used to classify the individual pixels. This involves the individual pixel error scores at the primary and secondary displacements.

The problem of detecting semantic events in video can be solved by a three-level approach as proposed in [4]. At the first level the input video sequence is decomposed into shots using a simple color histogram based technique, global motion is estimated, 76 low-level color and texture features are extracted, and motion blobs are detected. At the second level a multi-layer perceptron neural network is used to classify the detected motion blobs as moving object regions based on the extracted color and texture features. This level also summarizes each shot in terms of intermediate-level descriptors. At the third level the generated shot summaries are analyzed and the presence of the events of interest are detected based on an event inference model which incorporates domain-specific knowledge.

In this approach, the difference between the current and the motion compensated previous frame is used to detect motion blobs using a robust statistical estimation based algorithm [20]. Based on the frame difference result, the algorithm constructs two 1D histograms by projecting the frame difference map along its x and y direction, respectively. The histograms, therefore, represent the spatial distributions of the motion pixels along the corresponding axes. The instantaneous center position and size of a moving object in the video can be estimated based on statistical measurements derived from the two 1D projection histograms.

In [17], ASSET-2 (A Scene Segmenter Establishing Tracking), a motion segmentation and shape tracking system has been presented. The paper describes a way image sequences taken by a moving video camera may be processed to detect and track moving objects against a moving background in real-time.

In this approach, each frame is initially processed to find two dimensional features and edges. The two dimensional features are found using the SUSAN corner detector or the Harris corner detector. The SUSAN edge detector is used to find the edges. The two dimensional feature list is passed to a feature tracker which uses a two dimensional motion model to match and track features over as many frames as possible. A two dimensional vector field can then be created by taking either feature velocities or displacements over a fixed number of frames. The resulting vector field is passed to a flow segmenter which splits the list of flow vectors into clusters which have similar flow within them and are different to each other. Next, this cluster list is compared with a temporally filtered list of clusters, and the filtered list is updated using the newly found clusters. This list of clusters is finally used to provide information about the motions of objects.

In [20], an object matching system is described. It is able to extract objects of interest from outdoor scenes and match them. The application involves measuring the average travel time in a road network. Two cameras are placed about a mile apart from each other. Image analysis is performed on both the road image sequences to extract the objects of interest. An integration of multiple cues, including a motion segmentation mask of the moving areas in the image sequence, homogeneous color regions, edges, and model information is used to identify the moving objects. Two objects extracted from images captured by the two independent cameras at different times are then matched to evaluate their similarity.

The method to extract moving objects from an image sequence is based on the fusion of a motion segmentation mask obtained by image subtraction and a set of homogeneous color regions obtained by the integration of a split-and-merge algorithm and edges resulting from the Canny edge detector. Motion segmentation based on image subtraction is used to obtain an approximate location of the moving objects in the image sequence.

### **3 Video Object Tracking**

In order to find the moving objects we maintained a weighted average of the scene to be used as the background reference frame. Subtracting the current frame from the background reference frame results in blobs of moving objects. We described the blobs using an eigenvector/value decomposition of the region within the blob. A blob is thus represented by a vector of  $n$  values and  $n$  parameters; and the  $(x,y)$  origin parameters.

We applied an approach similar to [1] which used Kalman filtering (Appendix) to maintain the object identity over the video sequence and to optimize the tracking process of each blob. Figure 1 displays an example of blob segmentation where the left image is the original frame and the right image contains the segmented person.

The tracking method is important in that it is the first stage of segmentation. Any errors which appear in the tracking stage will also propagate to the classification stage described next.

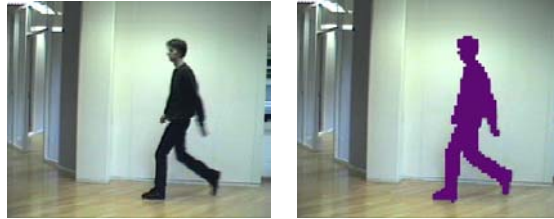


Figure 1. Frame from sequence (left); Segmented blob using background subtraction and Kalman Filtering (right).

#### 4 Classification

Understanding video sequences is a well researched topic (see [1], [5], [16], and [8]). The blobs are represented using their eigenvector decomposition [15]. We selected  $N$  eigenvectors based on maximizing the information content [5, 13, 22]. For the classification, we turned to the statistical literature for the probability ratio (PRT) test, which we use to classify the blobs as people, automobiles, or unknown.

- (1) Video segmentation and tracking similar to [1].
- (2) Blob region representation using eigenvector selection [5, 13, 15, 16, 22].
- (3) Initial blob region classification into 3 classes: people, automobiles, or unknown using PRT.

##### Median Classification and Standard Rate (SR)

The misdetection rate for the single frame classification method is defined as the standard rate (SR). As a certain object is tracked during a particular period of time, a certain history is built up. For example, if an object is tracked during eleven successive image frames, eleven classifications have been made, eleven classification values have been assigned. These classification values can be used to determine a global classification value for the whole sequence. The number of 0 classifications and the number of 1 classifications is determined. If the number of 0 classifications is greater than the number of 1 classifications, the global classification will be 0 (“no moving person”). If the number of 1 classifications is greater than the number of 0 classifications, the global classification will be 1 (“moving person”). If the number of 1 classifications equals the number of 0 classifications, the global classification will be 0.5 (“undecided”).

Instead of counting the different classifications, the average classification value of the sequence is computed. This is a faster way to determine the global classification value. If the average equals 0.5, then the classification will be “undecided”. If the average is less than 0.5, then the classification is “no moving person”. If the average is greater than 0.5, the classification will be “moving person”.

In order to reduce noise, to reduce the number of misclassifications in the SR sequence, a median filter can be used to determine new classification values. In this way, a smoothing of the classification sequence can be achieved.

Five methods, all based on median filtering, are developed. In the following sections, these methods are presented and evaluated. In these methods, the median filtering as presented is actually achieved by the average classification approach, as described above.

### Inside Filtering (IF)

A median filter can be seen as a rectangular window moving from the left to the right over the sequence of classification values. A median filter of size five for example will first take the first five values. The values are sorted and the middle value is taken to adjust the third value of the classification sequence as shown in Figure 2.

Then the filter is moved one place to the right and the process starts again. Five values are taken. The median of these five values will be assigned to the fourth value of the classification sequence. In this way, when using a filter of size five, the filter runs from the third element from the beginning of the sequence to the third element from the end of the sequence. We can define an inside and an outside. The outside elements are in this example the first two and the last two elements of the sequence.

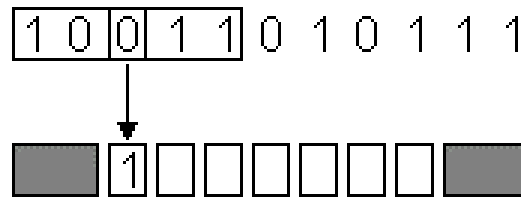


Figure 2. A median filter: Filter Size = 5.

As the method works on the inside of the sequence, it is called Inside Median Filtering (IF). The outside values remain the same.

### Outside Filtering (OF)

In this method, the filter used only works on the outside elements. The process is as follows. Imagine two virtual values before the first element and two virtual values after the last element of the classification sequence. The filter of size five can be used on the two virtual values and the first three values of the sequence. As we have only three values available, the median of these three values will be determined (see Figure 3). The second value will be based on the first four values.

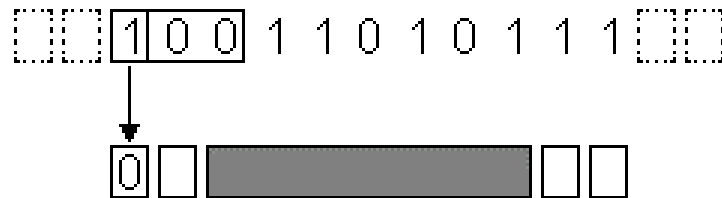


Figure 3. Outside Median Filtering: Filter Size = 5.

### Inside & Outside (IOF)

This method is a combination of the Inside and the Outside methods. These methods are applied at the same time. The filter starts assigning a value to the first element of the sequence. In our example, the Outside method will be used for the first two elements. After that, the Inside method is used seven times. Finally, the Outside method is used again to assign values to the last two elements of the sequence.

### InsideOutside (IFO)

A different combination is used here. First, the Inside method is used. This provides a new sequence of classification values. Then the Outside method is used on the new values.

### OutsideInside (OFI)

As in the previous method, only now the Outside method is used first.

## 5 Results

It is notable that our system uses off-the-shelf components which can be found in a typical computer/electronics store. We expect our system to be challenged by the following sources of noise:

- low resolution/detail images
- color distortion
- lens distortion
- loss of brightness and contrast from the video capture process.
- artifacts from the video tracking/segmentation process.
- block compression artifacts inherent in MPEG-1

In Figure 4, we display an example of the detection of automobiles and people in a busy street scene.



Figure 4. Example of detecting automobiles and people.

For our tests, we used 6 video sequences of city street intersections. Each sequence was 5 minutes (at 25 fps) in length and captured using a PAL camcorder. The video was extracted to 1.5 Mbps MPEG-1 digital format using an ATI All-In-Wonder Rage 128 card. The frame resolution was half PAL resolution.

Moreover, the system is expected to function in situations where the size of the moving object is quite small (less than 20 pixels wide).

On a PIII-800 Mhz computer, our system was able to capture, track, segment, and classify all of the blobs at a rate of 19 fps.

Figure 5 shows the results of using different filter sizes to improve the initial classification rate (SR).

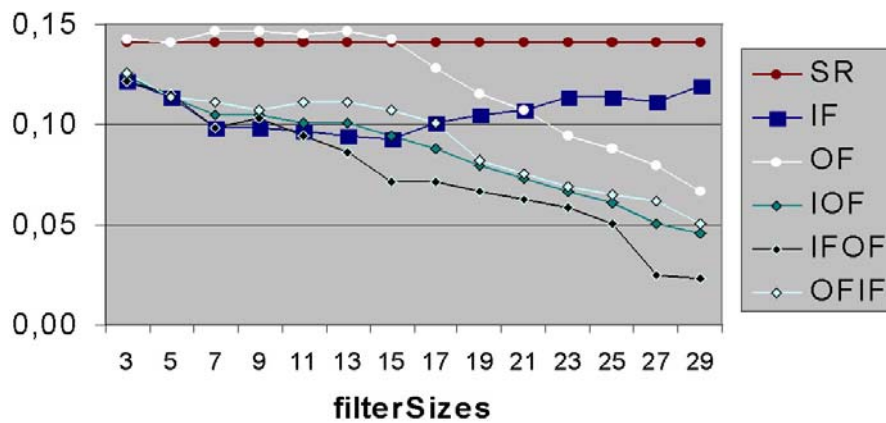


Figure 5. Misdetction (y-axis) vs. filter size (x-axis).

## 6 Discussion & Conclusions

In the previous sections we have described a system for recognizing two categories, people and automobiles from video sequences. The novel aspect of this paper is in applying several temporal averaging methods toward classifying the blobs. These methods exploit the sequential nature of video to improve the classification results.

In particular, each frame in the video containing the blob adds additional evidence toward the classification process. As more frames are processed, it can be shown that the error probabilities decrease.

From the tests, the IFOF method had the best misdetection rate. However, all of the median filter based methods improved the initial misdetection rate when sufficient frames were used.

Regarding future work, we think that significant improvement needs to be achieved in increasing the number of object categories and more abstract concepts. For example, it would be beneficial to classify abstract concepts such as a Christmas scene, or the emotional mood of the scene. Learning algorithms such as [5] and [8] appear to have the most potential. Therefore, methods based on techniques found in [2], [5], [7], [8], [15], and [22] are intended for exploration as well as matching in the compressed space [10]. Furthermore, the theoretical generalization of the PRT to sequential data, SPRT, will be investigated in later work.

## References

- [1] A.M. Baumberg, "Learning Deformable Models for Tracking Human Motion," PhD thesis, The University of Leeds, School of Computer Studies, UK, October 1995.
- [2] S.-F. Chang, W. Chen, H. J. Meng, H. Sundaram, D. Zhong, "VideoQ: An Automated Content Based Video Search System Using Visual Cues," *ACM Multimedia '97 Proceedings*, pp. 313-324, (Seattle, Washington, USA), November, 9-13, 1997.
- [3] H. Gu, Y. Shirai, M. Asada, "MDL-Based Segmentation and Motion Modeling in a Long Image Sequence of Scene with Multiple Independently Moving Objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 18, No. 1, pp. 58-64, January 1996.
- [4] M. Irani, P. Anandan, "A Unified Approach to Moving Object Detection in 2D and 3D Scenes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 20, No. 6, pp. 577-589, June 1998.
- [5] M. Lew, "Next Generation Web Searches for Visual Content," *IEEE Computer*, November, pp. 46-53, 2000.
- [6] H. S. Sawhney, S. Ayer, "Compact Representations of Videos Through Dominant and Multiple Motion Estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 18, No. 8, pp. 814-830, August 1996.

- [7] C. Stauffer, W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," *Proc. IEEE Computer Vision and Pattern Recognition*, (Fort Collins, Colorado), June 23-25, 1999.
- [8] M. Lew, T. Huang, K. Wong, "Learning and Feature Selection in Stereo Matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, September, pp. 869-881, 1994.
- [9] S. A. Niyogi, E. H. Adelson, "Analyzing and Recognizing Walking Figures in XYT," *Proc. IEEE Computer Vision and Pattern Recognition*, pp. 469-474, (Seattle, WA), June 21-23, 1994.
- [10] M. Lew, T. Huang, "Image Compression and Matching," *Proc. IEEE International Conference on Image Processing*, pp. 720-724, 1994.
- [11] M. Betke and N. Makris, "Fast Object Recognition in Noisy Images Using Simulated Annealing," *Proc. International Conference on Computer Vision*, pp. 523-530, June 1995.
- [12] M. Betke, E. Haritaoglu and L. S. Davis, "Real-Time Multiple Vehicle Detection and Tracking from a Moving Vehicle." *Machine Vision and Applications*, July 2000.
- [13] M. Lew, N. Sebe, "Visual Websearching Using Iconic Queries," *Proc. IEEE Computer Vision and Pattern Recognition*, pp 788-789, 2000.
- [14] Y. Ivanov and A. Bobick, "Recognition of Visual Activities and Interactions by Stochastic Parsing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, No. 8, pp. 852-872, August 2000.
- [15] A. Pentland, B. Moghaddam and T. Starner, "View-Based and Modular Eigenspaces for Face Recognition," *Proc. IEEE Computer Vision and Pattern Recognition*, 1994.
- [16] C. R. Wren, A. Azarbayejani, T. Darrell, A. P. Pentland, "Pfinder: Real-Time Tracking of the Human Body," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 19, No. 7, pp. 780-785, July 1997.
- [17] S. M. Smith, J. M. Brady, "ASSET-2: Real-Time Motion Segmentation and Shape Tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 8, pp. 814-820, August 1995.
- [18] I. Haritaoglu, D. Harwood and L. Davis, "W4: Real-Time Surveillance of People and Their Activities," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, No. 8, pp. 809-830, August 2000.
- [19] R. Zabih, J. Miller, K. Mai, "Video Browsing Using Edges and Motion," *Proc. IEEE Computer Vision and Pattern Recognition*, pp. 439-446, (San Francisco, California), June 18-20, 1996.
- [20] R. J. Qian, M. I. Sezan, K. E. Matthews, "Face Tracking Using Robust Statistical Estimation," *Proc. IEEE International Conference on Image Processing*, , vol. 1, pp. 131-135, 1998.
- [21] M.-P. Dubuisson, A. K. Jain, "2D Matching of 3D Moving Objects in Color Outdoor Scenes," *Proc. IEEE Computer Vision and Pattern Recognition*, pp. 887-891, (Seattle, WA), June 21-23, 1994.
- [22] M. Lew, D. Denteneer, "Fisher Keys for Content Based Retrieval," *Journal of Image & Vision Computing*, vol. 19, pp. 561-566, 2001.

## Appendix

### Kalman Filtering (see [1])

The Kalman equations used for the tracking were:

**Time update equations:**

$$\hat{x}_{k+1}(-) = A_k \hat{x}_k(+)$$
 (A.1)

$$P_{k+1}(-) = A_k P_k(+)$$
 (A.2)

**Measurement update equations:**

$$P_k^{-1}(+) = P_k^{-1}(-) + H_k^T R_k^{-1} H_k$$
 (A.3)

$$K_k = P_k(+)$$
 (A.4)

$$\hat{x}_k(+)$$
 (A.5)